University of Maryland
CMSC858K — Cryptography
Professor Jonathan Katz

Problem Set 2

Due at *beginning* of class on Mar. 11

1. **(Pseudorandom generators.)** Let $\{G_k : \{0,1\}^k \to \{0,1\}^{3k}\}$ and $\{H_k : \{0,1\}^k \to \{0,1\}^{3k}\}$ be PRGs. Prove (formally) or disprove (via explicit counterexample) whether the following are *necessarily* PRGs:

   (a) $\{G'_k : \{0,1\}^{2k} \to \{0,1\}^{3k}\}$ defined by:

   $$G'_k(x_1 \circ x_2) \stackrel{\text{def}}{=} G_k(x_1) \oplus G_k(x_2).$$

   (b) $\{H'_k : \{0,1\}^k \to \{0,1\}^{3k}\}$ defined by:

   $$H'_k(x) \stackrel{\text{def}}{=} G_k(x) \oplus H_k(x).$$

2. **(Pseudorandom functions.)** Let $\mathcal{F} = \{F_s : \{0,1\}^k \to \{0,1\}^k\}_{s \in \{0,1\}^k}$ be a PRF. Define $\mathcal{P} = \{P_s : \{0,1\}^{2k} \to \{0,1\}^{2k}\}_{s \in \{0,1\}^k}$ by:

   $$P_s(x_1 \circ x_2) \stackrel{\text{def}}{=} (F_s(x_1) \oplus x_2) \circ x_1.$$

   Iterating, define $\mathcal{P}' = \{P'_{s_1,s_2} : \{0,1\}^{2k} \to \{0,1\}^{2k}\}_{s_1,s_2 \in \{0,1\}^k}$ by:

   $$P'_{s_1,s_2}(x_1 \circ x_2) \stackrel{\text{def}}{=} P_{s_2}(P_{s_1}(x_1 \circ x_2)).$$

   (a) Write out a definition of $\mathcal{P}'$ in terms of $\mathcal{F}$ only.

   (b) (Review.) Show that $\mathcal{P}, \mathcal{P}'$ are *permutations* over their inputs.

   (c) (Review.) Show that, given s, P_s^{-1} can be efficiently computed (even if F_s^{-1} cannot). Repeat for $\mathcal{P}'$.

   (d) Show via explicit attack that $\mathcal{P}$ is *not* a pseudorandom permutation (PRP).

   (e) Show via explicit attack that $\mathcal{P}'$ is *not* a PRP.

   (f) Iterate the process a third time to define function family $\mathcal{P}''$. Write out your definition in terms of $\mathcal{F}$. Show that $\mathcal{P}''$ is *not* a *strong* PRP (we mentioned in class that $\mathcal{P}''$ *is* a PRP).

3. **(A PRP which is not a strong PRP.)** Given an efficiently invertible PRP $\mathcal{P} = \{P_s : \{0,1\}^k \to \{0,1\}^k\}_{s \in \{0,1\}^k}$ construct an *explicit* permutation family $\mathcal{P}'$ such that $\mathcal{P}'$ is a PRP but not a strong PRP. (You should be able to *prove* that your candidate $\mathcal{P}'$ is a PRP if $\mathcal{P}$ is, and you should show by explicit attack that $\mathcal{P}'$ is not a strong PRP. Make sure that $\mathcal{P}'$ is still an efficiently invertible permutation!)

4. **(Identification.)** Consider the following public-key identification scheme: the public key is a modulus N which is the product of two primes p, q such that $p = q = 3 \bmod 4$; the prover knows the factorization of N. Let $\mathcal{J}_N^{+1} \subset \mathbb{Z}_N^*$ denote those elements of $\mathbb{Z}_N^*$ with Jacobi symbol[1] $+1$. An execution of the scheme proceeds as follows: the verifier chooses a random $y \in \mathcal{J}_N^{+1}$ (this can be done efficiently, since the Jacobi symbol of $y \in \mathbb{Z}_N^*$ can be efficiently computed even without the factorization of N) and sends y as the challenge. The prover checks whether y or $-y$ is a quadratic residue (for N and y as above, exactly one of y or $-y$ is a quadratic residue), computes an arbitrary square root x for the appropriate one, and replies with x. The verifier checks whether $x^2 = \pm y \bmod N$.

   (a) Prove that this scheme is secure against a *passive* eavesdropper. In particular, show that an adversary who passively eavesdrops on multiple executions of the protocol and then impersonates the real prover can be used to factor N.

   (b) Prove that this scheme is *not* secure against an active adversary who may act as a verifier. In particular, show how an adversary acting as a dishonest verifier can recover the entire secret key.

[1] Note: you do not need to know anything about the Jacobi symbol in order to do this problem.